Scalable Hands Free Transfer Learning

Aug, 2014

Presented by: Brian d'Alessandro

Developed by: Brian d'Alessandro, Daizhuo Chen,
Troy Raeder, Melinda Han, Claudia Perlich, Foster Provost

Goal/Motivation

Goal:

Predict whether a person will convert after seeing an ad.

Estimate: P(Conv | Ad=1, X=x)

Goal/Motivation

Goal:
Predict whether a person will convert after seeing an ad.

Estimate: P(Conv | Ad=1, X=x)

How do you do this?
1. Serve ads randomly
2. Observe conversion
3. Build a data matrix
4. Train a model

UserID	Y	URL_1	URL_2	URL_3	…	URL_k
1	1	0	0	0	…	1
2	1	0	1	1	…	1
3	0	1	0	1	…	1
4	0	1	0	0	…	0
5	1	0	1	1	…	1
…	…	…	…	…	…	…
N	0	0	0	1	…	1

Goal/Motivation

So what's the problem?

Y is extremely sparse

K is large

UserID	Y	URL_1	URL_2	URL_3	...	URL_k
1	1	0	0	0	...	1
2	1	0	1	1	...	1
3	0	1	0	1	...	1
4	0	1	0	0	...	0
5	1	0	1	1	...	1
...	...	...	...	...	...	...
N	0	0	0	1	...	1

A large N is expensive

dstillery

Goals/Motivation

And there's more…

- Cold start

- Need to support many models

- Can't pool data across advertisers

Our Solution

Bayesian Transfer Learning with Adaptive Stochastic Gradient Descent

Bayesian Transfer Learning – Modify standard L1/L2 regularization with an informative prior to transfer model parameters learned from one problem to another.

Adaptive Stochastic Gradient Descent – Combine state-of-the-art in adaptive learning rates and adaptive regularization for hyper-parameter free learning.

Bayesian Transfer Learning

Two Sources of Data

We collect data via different data streams…

UserID	Y	URL_1	URL_2	URL_3	…	URL_k
1	1	0	0	0	…	1
2	1	0	1	1	…	1
3	0	1	0	1	…	1
4	0	1	0	0	…	0
5	1	0	1	1	…	1
…	…	…	…	…	…	…
N	0	0	0	1	…	1

Ad Serving (Target Data)

$$$$$$$$$$$$$

dstillery

Two Sources of Data

We collect data via different data streams…

UserID	Y	URL_1	URL_2	URL_3	…	URL_k
1	1	0	0	0	…	1
2	1	0	1	1	…	1
3	0	1	0	1	…	1
4	0	1	0	0	…	0
5	1	0	1	1	…	1
…	…	…	…	…	…	…
N	0	0	0	1	…	1

Ad Serving (Target Data)

$$$$$$$$$$$$

UserID	Y	URL_1	URL_2	URL_3	…	URL_k
1	0	1	0	1	…	1
2	0	0	1	0	…	1
3	0	1	1	1	…	0
4	1	0	1	1	…	0
5	1	1	1	0	…	0
…	…	…	…	…	…	…
N	0	1	1	0	…	0

General Web Browsing (Source Data)

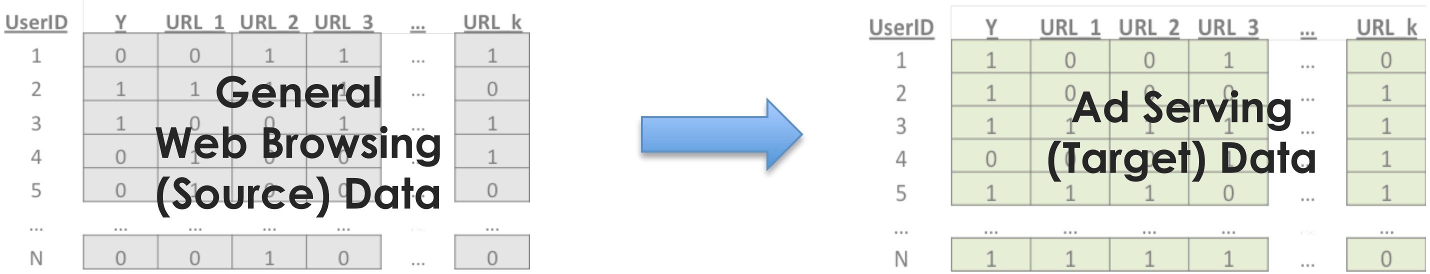
$

dstillery

Transfer Learning

Assume $f^S(X) \approx f^T(X)$ *and* $f^T(X)$ *is high variance*

Solution: *Use* $f^S(X)$ *as a regularization prior for* $f^T(X)$



Intuition: The auxiliary model is biased but much more reliable. Use this to inform the real model. The algorithm can learn how much of the auxiliary data to use.

dstillery

2 Stage Model Training

1. *Run a logistic regression on source data (using your favorite methodology)*

$$\hat{\mu} = \underset{\mu \in R^k}{\operatorname{argmin}} \; SourceLL(\mu)$$

2. *Use results of step 1 as informative-prior for Target model*

$$\hat{\beta} = \underset{\beta \in R^k}{\operatorname{argmin}} \; TargetLL(\beta)$$

$$LL = -\sum_{i=1}^{N} y * log(p) + (1 - y) * log(1 - p) - c * \sum_{j=1}^{k} (\beta - \hat{\mu})^2$$

dstillery

Scalable SGD

Nearly Hyper-Parameter Free

No More Pesky Learning Rates

Tom Schaul SCHAUL@CIMS.NYU.EDU
Sixin Zhang ZSX@CIMS.NYU.EDU
Yann LeCun YANN@CIMS.NYU.EDU
Courant Institute of Mathematical Sciences
New York University
715 Broadway, New York, NY 10003, USA

Learning Recommender Systems with Adaptive Regularization

Steffen Rendle
Social Network Analysis
University of Konstanz
78457 Konstanz, Germany
steffen.rendle@uni-konstanz.de

Both are approximate optimization problems

1. Adaptive Learning Rate: Find the learning rate that optimizes next training update

$$\eta_{t,i} \leftarrow \arg\min_{\eta} E_{\nabla_{t,i}}[f(\beta_{t,i} - \eta \nabla_{t,i} e_i)] \quad \longrightarrow \quad \eta_{t,i} = \frac{1}{h_{t,i}} \frac{g_{t,i}^2}{v_{t,i}}$$

2. Adaptive Regularization: At each training update, find the regularization weight that optimizes test dataset

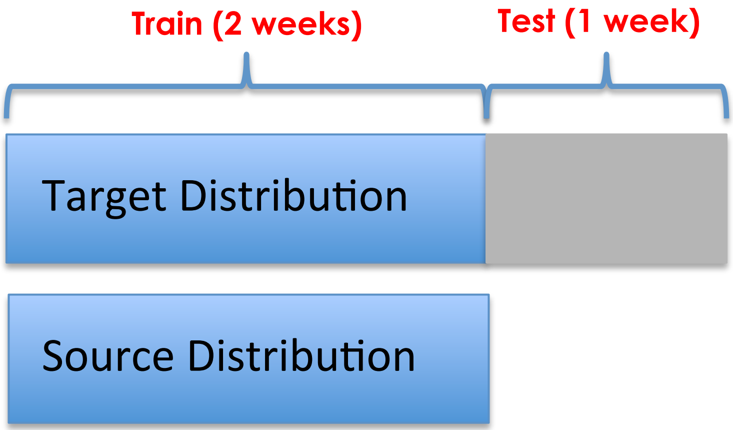
$$\lambda^* | \Theta^t := \underset{\lambda \in \mathbb{R}_+^c}{\operatorname{argmin}} \sum_{(\mathbf{x},y) \in S_V} l\left(\hat{y}(\mathbf{x}|\Theta^{t+1}), y\right) \quad \longrightarrow \quad \lambda^{t+1} = \lambda^t - \alpha \frac{\partial}{\partial \lambda} l\left(\hat{y}(\mathbf{x}|\Theta^{t+1}), y\right)$$

dstillery

Experiments

Experimental Set Up

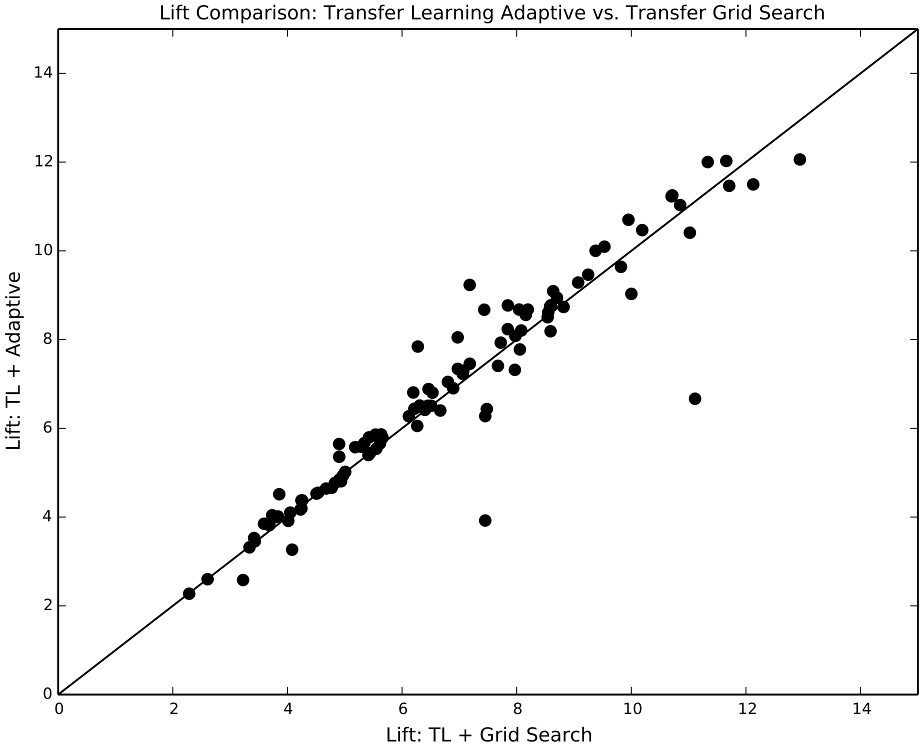
Data from 100 Dstillery Campaigns



Experimental Design

1. Transfer vs. No Transfer, traditional Grid Search
2. Grid Search vs. Adaptive Learning (with Transfer)
3. Adaptive Transfer vs. No Transfer with Grid Search

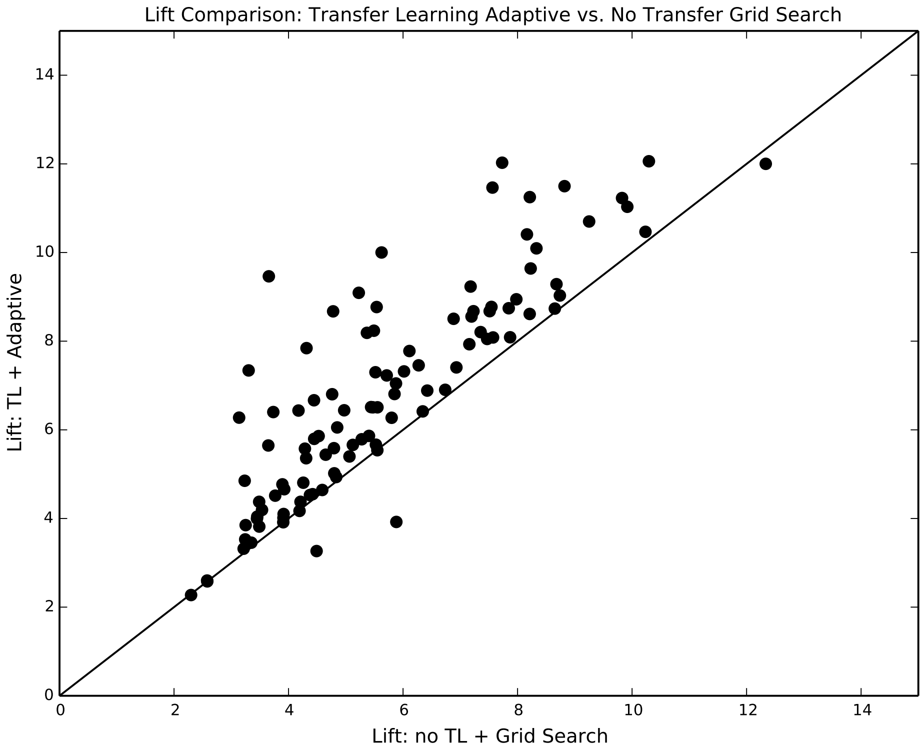
Grid Search vs. Adaptive Learning



Adaptive Learning:

- Results in no statistical difference in performance
- Is an order of magnitude faster

New Method vs. Default



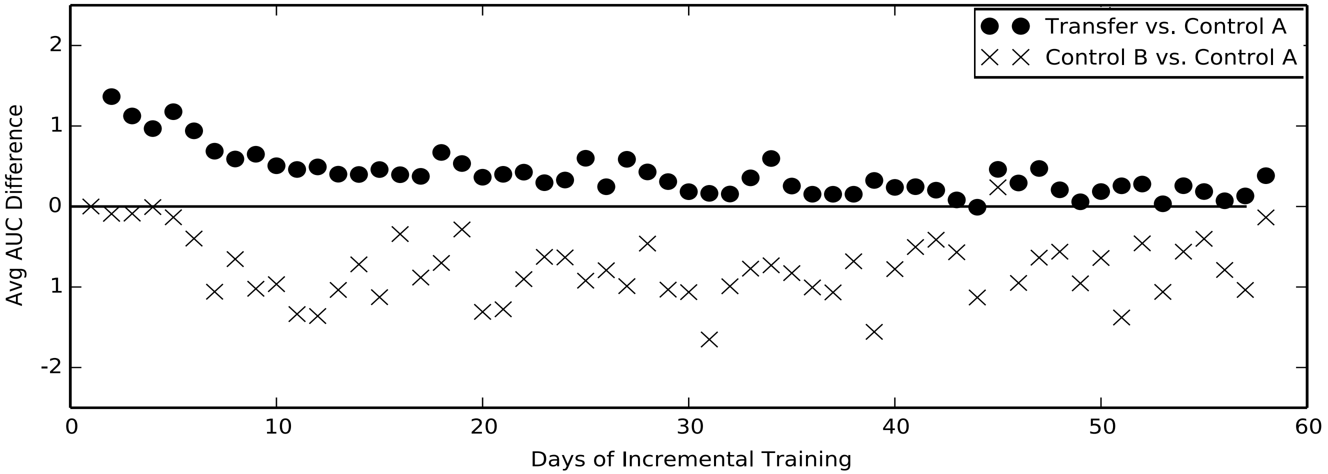
Transfer Learning w/ Adaptive Learning:

- Increased lift for 93% of tests
- Resulted in avg. Lift increase of 23%
- Is an order of magnitude faster than grid search

dstillery

Incremental Learning

Q: What happens over time as target task is able to use more data?

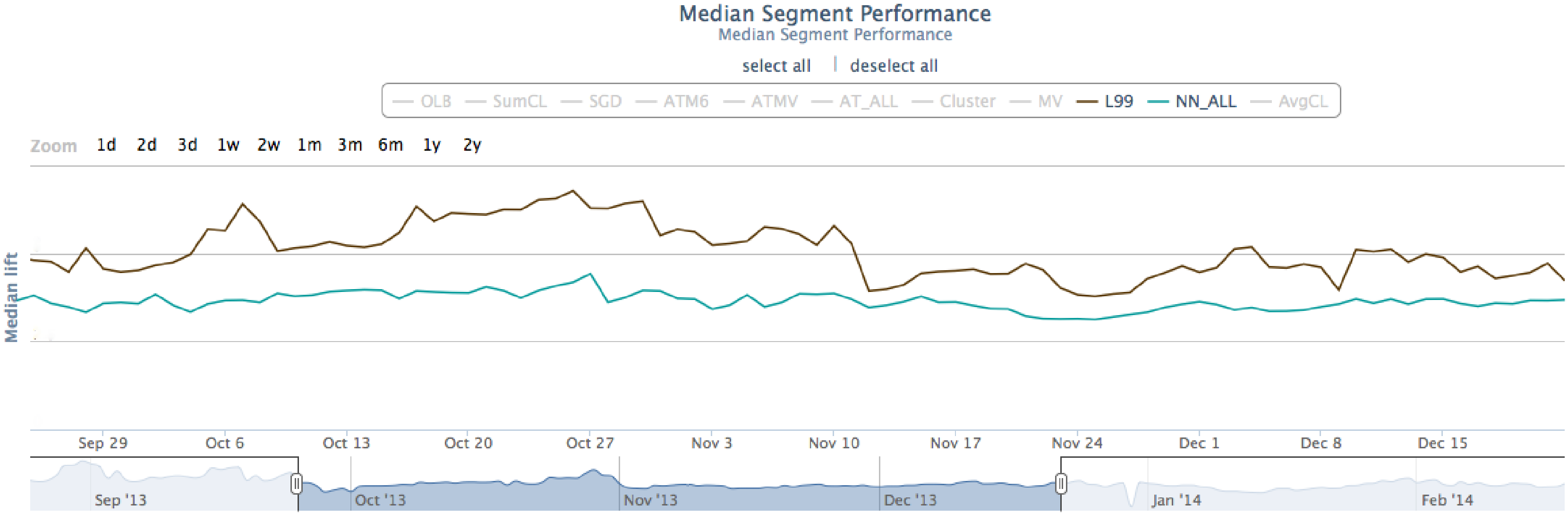


A: Transfer learning benefit degrades as campaign matures

dstillery

And it Works Live!

The approaches presented here generally dominate our baseline methods.



dstillery